

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

- 1 1. (Currently amended) A method for logging file system operations,
2 comprising:
3 receiving a request to perform a file system operation at a primary server;
4 making a call to an underlying file system to perform the file system
5 operation; and
6 logging the file system operation to a log within a log device to facilitate
7 recovery of the file system operation in the event of a system failure before the
8 file system operation is committed to non-volatile storage, wherein the log device
9 is located on a secondary server that is separate from the primary server, and
10 wherein the secondary server acts as a backup for the primary server;
11 wherein the file system operation includes arguments and data
12 needed to repeat the file system operation; and
13 wherein locating the log on the secondary server facilitates failover
14 to the secondary server when the primary server fails; and
15 wherein locating the log in volatile memory on the secondary
16 server facilitates recovery of the file system operation without adding delay to
17 normal file system operations due to writes to non-volatile storage;
18 periodically committing the log to the underlying file system in non-
19 volatile storage, and removing outstanding file system operations from the log in
20 volatile memory;
21 and

22 wherein upon a subsequent computer system startup, the method further
23 comprises:
24 examining the log within the log device; and
25 replaying any file system operations from the log;
26 wherein the log is only in volatile memory that are in volatile memory and
27 that have not been committed to non-volatile storage.

1 2. (Original) The method of claim 1, wherein logging the file system
2 operation involves storing an identifier for the file system operation to the log
3 device.

1 3. (Currently amended) The method of claim 1, ~~further comprising~~
2 wherein periodically committing the log to the underlying file system ~~by further~~
3 comprises:
4 freezing ongoing activity on a file system;
5 making a call to the underlying file system to flush memory buffers to
6 non-volatile storage, whereby outstanding file system operations are guaranteed
7 to be committed to non-volatile storage;
8 ~~removing outstanding file system operations from the log; and~~
9 unfreezing the ongoing activity on the file system.

1 4 (Canceled).

1 5. (Original) The method of claim 1, further comprising checking for
2 dependencies between the file system operation and ongoing file system
3 operations; and

4 if dependencies are detected, ensuring that the file system operation and
5 the ongoing file system operations complete in an order that satisfies the
6 dependencies.

1 6 (Canceled).

1 7. (Original) The method of claim 1, further comprising:
2 associating the file system operation with a transaction identifier for a set
3 of related file system operations; and
4 wherein logging the file system operation involves storing the file system
5 operation with the transaction identifier to the log device.

1 8. (Original) The method of claim 1, wherein logging the file system
2 operation involves:
3 determining if the file system operation belongs to a subset of file system
4 operations that are subject to logging; and
5 if so, logging the file system operation.

1 9. (Original) The method of claim 8, wherein the subset of file system
2 operations are non-idempotent file system operations.

1 10. (Original) The method of claim 1, wherein the log device stores the
2 file system operation in volatile storage.

1 11. (Original) The method of claim 1, wherein the log device stores the
2 file system operation in non-volatile storage.

1 12. (Currently amended) A computer-readable storage medium storing
2 instructions that when executed by a computer cause the computer to perform a
3 method for logging file system operations, wherein the computer-readable storage
4 medium includes one of a volatile memory, a non-volatile memory, a disk drive, a
5 magnetic tape, a compact disc, a digital versatile disc, and a digital video disk, the
6 method comprising:
7 receiving a request to perform a file system operation at a primary server
8 in a highly available system;
9 making a call to an underlying file system to perform the file system
10 operation; and
11 logging the file system operation to a log within a log device to facilitate
12 recovery of the file system operation in the event of a system failure before the
13 file system operation is committed to non-volatile storage, wherein the log device
14 is located on a secondary server that is separate from the primary server in the
15 highly available system and wherein the secondary server acts as a backup for the
16 primary server;
17 wherein the file system operation includes arguments and data needed to
18 repeat the file system operation; and
19 wherein locating the log on the secondary server facilitates failover to the
20 secondary server when the primary server fails; and
21 wherein locating the log in volatile memory on the secondary server
22 facilitates recovery of the file system operation without adding delay to normal
23 file system operations due to writes to non-volatile storage;
24 periodically committing the log to the underlying file system in non-
25 volatile storage, and removing outstanding file system operations from the log in
26 volatile memory;
27 and

28 wherein upon a subsequent computer system startup, the method further
29 comprises:
30 examining the log within the log device; and
31 replaying any file system operations from the log, ~~wherein the log~~
32 is only in volatile memory that are in volatile memory and that have not been
33 committed to non-volatile storage.

1 13. (Original) The computer-readable storage medium of claim 12,
2 wherein logging the file system operation involves storing an identifier for the file
3 system operation to the log device.

1 14. (Currently amended) The computer-readable storage medium of claim
2 12, wherein ~~the method further comprises~~ periodically committing the log to the
3 underlying file system by further comprises:
4 freezing ongoing activity on a file system;
5 making a call to the underlying file system to flush memory buffers to
6 non-volatile storage, whereby outstanding file system operations are guaranteed
7 to be committed to non-volatile storage;
8 ~~removing outstanding file system operations from the log;~~ and
9 unfreezing the ongoing activity on the file system.

1 15 (Canceled).

1 16. (Original) The computer-readable storage medium of claim 12,
2 wherein the method further comprises checking for dependencies between the file
3 system operation and ongoing file system operations; and

4 if dependencies are detected, ensuring that the file system operation and
5 the ongoing file system operations complete in an order that satisfies the
6 dependencies.

1 17 (Canceled).

1 18. (Original) The computer-readable storage medium of claim 12,
2 wherein the method further comprises:
3 associating the file system operation with a transaction identifier for a set
4 of related file system operations; and
5 wherein logging the file system operation involves storing the file system
6 operation with the transaction identifier to the log device.

1 19. (Original) The computer-readable storage medium of claim 12,
2 wherein logging the file system operation involves:
3 determining if the file system operation belongs to a subset of file system
4 operations that are subject to logging; and
5 if so, logging the file system operation.

1 20. (Original) The computer-readable storage medium of claim 19,
2 wherein the subset of file system operations are non-idempotent file system
3 operations.

1 21. (Original) The computer-readable storage medium of claim 12,
2 wherein the log device stores the file system operation in volatile storage.

1 22. (Original) The computer-readable storage medium of claim 12,
2 wherein the log device stores the file system operation in non-volatile storage.

1 23. (Currently amended) An apparatus that logs file system operations,
2 comprising:
3 a receiving mechanism that is configured to receive a request to perform a
4 file system operation at a primary server in a highly available system;
5 a calling mechanism that is configured to make a call to an underlying file
6 system to perform the file system operation; and
7 a logging mechanism that is configured to log the file system operation to
8 a log within a log device to facilitate recovery of the file system operation in the
9 event of a system failure before the file system operation is committed to non-
10 volatile storage, wherein the log device is located on a secondary server that is
11 separate from the primary server in the highly available system and wherein the
12 secondary server acts as a backup for the primary server;
13 wherein the file system operation includes arguments and data
14 needed to repeat the file system operation; and
15 wherein locating the log on the secondary server facilitates failover
16 to the secondary server when the primary server fails; and
17 wherein locating the log in volatile memory on the secondary
18 server facilitates recovery of the file system operation without adding delay to
19 normal file system operations due to writes to non-volatile storage;
20 a committing mechanism for periodically committing the log to the
21 underlying file system in non-volatile storage, and removing outstanding file
22 system operations from the log in volatile memory;
23 and
24 a recovery mechanism that operates during system startup, wherein the
25 recovery mechanism is configured to:
26 examine the log within the log device; and to

27 replay any file system operations from the log, ~~wherein the log is~~
28 ~~only in volatile memory~~ that are in volatile memory and that have not been
29 committed to non-volatile storage.

1 24. (Original) The apparatus of claim 23, wherein the logging mechanism
2 is configured to store an identifier for the file system operation to the log device.

1 25. (Currently amended) The apparatus of claim 23, wherein ~~the logging~~
2 ~~mechanism is configured to periodically~~ the committing mechanism further
3 comprises:
4 ~~freeze-freezing~~ ongoing activity on a file system;
5 ~~make-making~~ a call to the underlying file system to flush memory buffers
6 to non-volatile storage, whereby outstanding file system operations are
7 guaranteed to be committed to non-volatile storage;
8 ~~remove-outstanding file system operations from the log; and to~~
9 ~~unfreeze-unfreezing~~ the ongoing activity on the file system.

1 26 (Canceled).

1 27. (Original) The apparatus of claim 23, further comprising a dependency
2 handler that is configured to:

3 check for dependencies between the file system operation and ongoing file
4 system operations; and to

5 ensure that the file system operation and the ongoing file system
6 operations complete in an order that satisfies dependencies if dependencies are
7 detected.

1 28 (Canceled).

1 29. (Original) The apparatus of claim 23, further comprising a transaction
2 mechanism that is configured to associate the file system operation with a
3 transaction identifier for a set of related file system operations; and
4 wherein the logging mechanism is configured to log the file system
5 operation with the transaction identifier to the log device.

1 30. (Original) The apparatus of claim 23, wherein the logging mechanism
2 is configured to:
3 determine if the file system operation belongs to a subset of file system
4 operations that are subject to logging; and to
5 log the file system operation if the file system operation belongs to the
6 subset of file system operations that are subject to logging.

1 31. (Original) The apparatus of claim 30, wherein the subset of file system
2 operations are non-idempotent file system operations.

1 32. (Original) The apparatus of claim 23, wherein the log device is
2 configured to store the file system operation in volatile storage.

1 33. (Original) The apparatus of claim 23, wherein the log device is
2 configured to store the file system operation in non-volatile storage.

1 34. (Previously presented) The method of claim 1, wherein the primary
2 server is in a highly available system; and wherein the secondary server is in the
3 highly available system.

1 35. (Previously presented) The computer-readable storage medium of
2 claim 12, wherein the primary server is in a highly available system; and wherein
3 the secondary server is in the highly available system.

1 36. (Previously presented) The apparatus of claim 23, wherein the primary
2 server is in a highly available system; and wherein the secondary server in the
3 highly available system.